CSCI2100/ESTR2102: Quiz 2

Name: Student ID:

Problem 1 (15%). After applying the following operations to an empty stack:

push(35), push(36), push(43), push(8), pop, pop, push(51), pop

what is the content of the stack?

Solution. 35, 36

Problem 2 (15%). After applying the following operations to an empty queue:

enqueue(35), enqueue(36), enqueue(43), enqueue(8), dequeue, dequeue, enqueue(51), dequeue

what is the content of the queue?

Solution. 8, 51.

Problem 3 (15%). Indicate whether the following statements are correct. If you think the statement is **incorrect**, you need to **explain why.**

1. Consider a data structure that supports a certain operation in $O(1)$ amortized time. Then, any sequence of n such operations requires $O(n)$ worst case time, when n is larger than a certain constant.
2. Consider a data structure that supports a certain operation in $O(1)$ amortized time. But still, it is possible for the structure to take $\Omega(n)$ time to process *one* operation, where n is the number of operations that have already been processed.

Solution. Both statements are correct.

Problem 4 (25%). Consider the hash function $h(k) = 1 + k \bmod 7$. Give a set S of 10 integers to meet both conditions below:

- If we build a hash table on S using $h(k)$, then all the integers of S fall in the same bucket (recall that a *bucket* contains all the elements of S having the same hash value).
- The aforementioned bucket is the one we probe in order to look for integer 35.

Solution. $S = \{7i \mid i = 1, 2, ..., 10\}$.

Problem 5 (30%). You have a linked list that stores a set S of n integers. Given a search value q, we want to remove from the linked list the predecessor of q in S. Describe an algorithm that does so in $O(n)$ time.

Solution. Scan the linked list from head to tail. Among all the integers encountered, maintain the largest one that is less than or equal to q. At the end of the scan, we have the node storing the predecessor of q. Remove the node from the linked list.